Optimization for Gaussian Processes via Chaining

Emile Contal **Cédric Malherbe** **Nicolas Vayatis**
CMLA, ENS Cachan, CNRS, Université Paris-Saclay, 94235, Cachan, France
contal@cmla.ens-cachan.fr

Abstract

In this paper, we consider the problem of stochastic optimization under a bandit feedback model. We generalize the GP-UCB algorithm [Srinivas and al., 2012] to arbitrary kernels and search spaces. To do so, we use a notion of localized chaining to control the supremum of a Gaussian process, and provide a novel optimization scheme based on the computation of covering numbers. The theoretical bounds we obtain on the cumulative regret are more generic and present the same convergence rates as the GP-UCB algorithm. Finally, the algorithm is shown to be empirically more efficient than its natural competitors on simple and complex input spaces.

1 Introduction

Optimizing an unknown and noisy function is at the center of many applications in the field of machine learning [1]. The goal of a sequential optimization procedure may be either seen as maximizing the sum of the outputs (or rewards) received at each iteration, that is to minimize the cumulative regret widely used in bandit problems, or as maximizing the best reward received so far, that is to minimize the simple regret. This task becomes challenging when we only have mild information about the unknown function. To tackle this challenge, a Bayesian approach has been shown to be empirically efficient [2, 3, 4, 5, 6]. In this approach, we model the unknown function as a realization of a Gaussian Process (GP) which allows to control the assumptions we put on the smoothness of the function by choosing different kernels [7]. In order to prove theoretical guarantees on the regret, algorithms in the literature typically rely on high probabilistic upper confidence bounds (UCB) [8, 9, 10, 11, 12]. In these works and many others, the UCB is obtained with a union bound over all the points of a discretization of the input space. The major drawback of this approach is that the UCB depends on the cardinality of the discretization instead of the complexity of the input space itself. As a consequence, the convergence rates derived on the regret become arbitrary large when the discretization becomes finer and finer. Aiming at filling this gap between theory and practice we propose an efficient computation of chaining for Bayesian Optimization. Chaining, that has been recently studied in the context of sequential optimization (see [13] for bandit with known horizon or [14] in the case of online regression), appears to be an ideal tool to capture the complexity of the search space with respect to the smoothness of the function. The contribution of this paper is twofold: we first introduce a novel policy based on the computation of covering numbers called the CHAINING-UCB that can be seen as a generalization of the GP-UCB algorithm with automatic calibration of the exploration/exploitation tradeoff for arbitrary kernels and search space. On the other hand, we provide theoretical guarantees on its regret with the same convergence rates as its competitors, without depending on the cardinality of the discretization of the search space. The rest of the paper is organized as follows: in Section 2, we present the framework of our analysis, the basic properties of GP and we introduce the CHAINING-UCB algorithm. In Section 3, we present and discuss the upper bound on the regret. Finally in Section 4 we compare the empirical performances of the CHAINING-UCB algorithm to its natural competitors on simple input spaces, that is $\mathbb{R}^D$, and complex input spaces, this is directed graphs space.

Algorithm 1: CHAINING-UCB($\mathcal{X}, k(\cdot,\cdot), \eta, \delta$)

for $t = 1, 2, \dots$ **do**
  Compute μ_t, σ_t and d_t
  $T_0 \leftarrow \emptyset$; $\sigma_t^{\min} = \min_{x\in\mathcal{X}} \sigma_t(x)$
  for $i = 1 \dots \lceil 1 - \log_2(\sigma_t^{min}) \rceil$ **do**
    $\epsilon_i \leftarrow 2^{-i+1}$
    $\bar{\mathcal{X}} \leftarrow \{x \in \mathcal{X} : d_t(x, T_{i-1}) > \epsilon_i\}$
    $T_i \leftarrow T_{i-1} \cup$ COVER($\bar{\mathcal{X}}, d_t, \epsilon_i$)
    $H_i \leftarrow \epsilon_i \sqrt{2\log\left((|T_i|+1) i^2 t^2 \frac{\pi^4}{\delta 6^2}\right)}$
  end
  $x_t \leftarrow \operatorname*{argmax}_{x\in\mathcal{X}} \mu_t(x) + \sum_{i:\sigma_t^{\min} \le \epsilon_i < \sigma_t(x)} H_i$
  Sample x_t and observe y_t
end

Algorithm 2: GREEDY-COVER($\mathcal{X}, d, \epsilon$)

$T \leftarrow \emptyset$; $\bar{\mathcal{X}} \leftarrow \mathcal{X}$
$\forall x, x' \in \mathcal{X},\ G[x,x'] \leftarrow \mathbb{1}_{d(x,x')\le\epsilon}$
while $\bar{\mathcal{X}} \neq \emptyset$ **do**
  $x \leftarrow \operatorname{argmax}_{x\in\bar{\mathcal{X}}} \sum_{x'\in\bar{\mathcal{X}}} G[x,x']$
  $T \leftarrow T \cup \{x\}$
  $\bar{\mathcal{X}} \leftarrow \bar{\mathcal{X}} \setminus \{x' \in \bar{\mathcal{X}} : G[x,x'] = 1\}$
end
return T

2 The CHAINING-UCB algorithm

Bayesian optimization framework. Let $f : \mathcal{X} \to \mathbb{R}$ be the unknown function we want to optimize, where $\mathcal{X}$ is the input space which is not necessarily a subset of $\mathbb{R}^D$. We assume that f is a realization of a centered Gaussian process with known kernel k satisfying $k(x,x') \le 1$ for all $(x,x') \in \mathcal{X}^2$. To avoid measurability issues we suppose that $\mathcal{X}$ is a finite set with arbitrary cardinality (see [15, 16] for more details), and we do not address the computational problem of handling continuous spaces. A sequential optimization algorithm iterates two steps: it first chooses x_n based on $y_1, \dots, y_{n-1}$, and next gets the noisy observation $y_n = f(x_n) + \epsilon_n$ where $(\epsilon_n)_{n\ge1}$ are independent Gaussians $\mathcal{N}(0, \eta^2)$ with known variance η^2. Let $X_n = \{x_1, \dots, x_n\}$ be the set of queried points after n iterations and $\mathbf{Y}_n = [y_1, \dots, y_n]$ the associated observations packed in vector form. Unlike the work of [13] this paper considers that the time horizon n is unknown. For GP, the distribution of f conditioned on $\mathbf{Y}_n$ is a non-centered GP of mean μ_{n+1} and kernel k_{n+1} computed as follows for all $(x,x') \in \mathcal{X}^2$:

$$\mu_{n+1}(x) = \mathbf{k}_n(x)^\top \mathbf{C}_n^{-1} \mathbf{Y}_n \ \text{ and } \ k_{n+1}(x,x') = k(x,x') - \mathbf{k}_n(x)^\top \mathbf{C}_n^{-1} \mathbf{k}_n(x')\,, \tag{1}$$

where $\mathbf{k}_n(x) = [k(x_t, x)]_{x_t\in X_n}$ is the kernel vector between x and X_n, and $\mathbf{C}_n = \mathbf{K}_n + \eta^2 \mathbf{I}$ with $\mathbf{K}_n = [k(x_t, x_{t'})]_{x_t, x_{t'} \in X_n}$ the kernel matrix [7]. We also define the variance $\sigma_n^2(x) = k_n(x,x)$ and the pseudo-distance $d_n(x, x^\star)$:

$$d_n(x, x^\star) = \sqrt{\sigma_n^2(x^\star) - 2k_n(x, x^\star) + \sigma_n^2(x)}\,. \tag{2}$$

Note that $d_n^2(x, x^\star) = \operatorname{Var}[f(x^\star) - f(x) \,|\, X_n, \mathbf{Y}_n]$. To measure the complexity of $\mathcal{X}$ we will compute covering numbers with respect to d_n.

An upper confidence bound algorithm via chaining. At the core of our strategy to control the regret of the algorithm we need an upper confidence bound (UCB) on $\sup_{x^\star\in\mathcal{X}} f(x^\star) - f(x)$ for all $x \in \mathcal{X}$. A naive approach uses a union bound on $\mathcal{X}$, resulting in a factor $\sqrt{\log|\mathcal{X}|}$ in the UCB, which is not appropriate when $\mathcal{X}$ is a numerical discretization of a continuous space, typically a multidimensional grid of high density. We use the chaining trick [17, 18, 19] to get a UCB relying on the covering numbers of $\mathcal{X}$ with respect to d_n instead of its cardinality. In that way the algorithm adapts to arbitrary input spaces. The main element of our algorithm is the computation of hierarchical ϵ-covers of $\mathcal{X}$. We say that a subset T is an ϵ-cover of $\mathcal{X}$ when for all $x \in \mathcal{X}$, $d_n(x, T) \le \epsilon$, where $d_n(x, T) = \inf_{x'\in T} d_n(x, x')$. The covering numbers $N(\mathcal{X}, d_n, \epsilon)$ are the cardinality of the smallest ϵ-cover of $\mathcal{X}$ for the pseudo-distance d_n, and the function COVER($\mathcal{X}, d_n, \epsilon$) in Algorithm 1 returns such a set. The CHAINING-UCB algorithm then queries the objective function at the point maximizing the UCB obtained by chaining. The computation of an optimal ϵ-cover is NP-hard, but we can easily build an efficient approximation as shown in Algorithm 2 and discussed in Section 4.

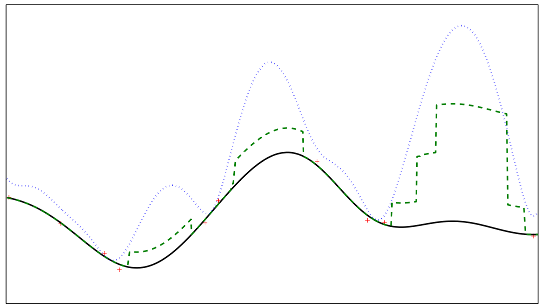


Figure 1: Illustration of the exploration/exploitation tradeoff maximized in Algorithm 1. The red crosses are the noisy observations. The plain black line is the posterior mean μ. The dashed green line is the target of the CHAINING-UCB algorithm. The dotted blue line is the target used by the GP-UCB algorithm. Remark that the rectangular form is explained by the discrete sum.

3 Theoretical analysis

Guarantees on the regret. In the following theorem we provide a high probabilistic upper bound on the instantaneous regret incurred by Algorithm 1 in terms of the posterior deviations $\sigma_n(x_n)$ and covering numbers. This inequality is used in the subsequent corollary to obtain upper bounds on its cumulative and simple regrets.

Theorem 1. *For any finite $\mathcal{X}$, let $x_1, x_2, \dots$ be the queries of the* CHAINING-UCB *algorithm on f sampled from a $\mathcal{GP}(0,k)$ where $k(\cdot,\cdot) \leq 1$. For $\delta \in (0,1)$, using the notations $\sigma_n = \sigma_n(x_n)$ and $c_{n,\delta} = 6\sqrt{\log \frac{n^2\pi^4}{36\delta}} + 15$, we have with probability at least $1-\delta$ that for all $n \in \mathbb{N}^\star$:*

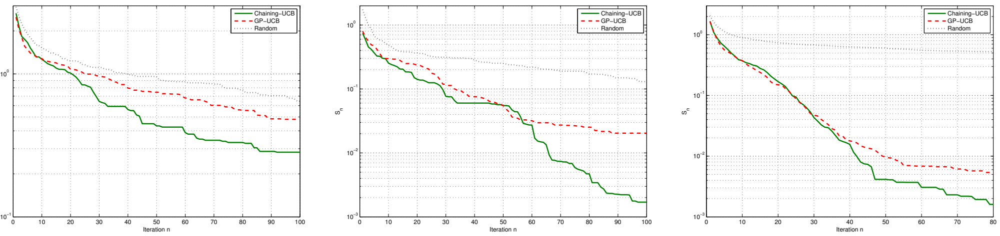
$$\sup_{x^\star \in \mathcal{X}} f(x^\star) - f(x_n) \leq \sigma_n\big(c_{n,\delta} - 6\log\sigma_n\big) + 9 \sum_{i:2^{-i}<\sigma_n} 2^{-i}\sqrt{\log N(\mathcal{X}, d_n, 2^{-i})}\,.$$

In order to simplify this inequality and get convergence rates for the cumulative regret R_n and simple regret S_n, it is necessary to add some assumptions on k and $\mathcal{X}$. Corollary 1 gives an example of the rates we obtain for the usual Squared-Exponential kernel $k(x,x') = e^{-\frac{1}{2}\lVert x-x' \rVert_2^2}$ and $\mathcal{X}$ in $\mathbb{R}^D$.

Corollary 1. *For the SE kernel and a compact $\mathcal{X} \subseteq [0,R]^D$, the* CHAINING-UCB *algorithm incurs regrets $R_n = \mathcal{O}\Big(\sqrt{n(\log n)^{D+2}}\Big)$ and $S_n = \mathcal{O}\Big(\sqrt{\frac{(\log n)^{D+2}}{n}}\Big)$ with probability at least $1-\delta$.*

The proof of Theorem 1 employs the chaining trick to get a local control on the supremum of a non-centered GP. Since this requires to define additional structures, the proofs are not included in the present article and we refer instead to the technical paper [20]. The proof of Corollary 1 first uses an upper bound on the covering numbers via the Lipschitz property of the SE kernel. It then applies the information-theoretical inequality proven in Theorem 5 of [9] to obtain the given regret rates. It is straightforward to apply this technique to other cases like linear kernels or Matérn kernels with parameter $\nu > 2$, since the covering numbers can be bounded by similar techniques.

A flexible algorithm. A reader familiar with classical chaining may ask why we use the bad-looking sum $\sum_{i:\epsilon_i<\sigma_n} \epsilon_i \sqrt{\log N(\mathcal{X}, d_n, \epsilon_i)}$ instead of the Dudley integral. Even if the Dudley integral is simple to bound for certain kernel k and space $\mathcal{X}$, we want Algorithm 1 to be able to adapt to all search space without having to tune its parameters. By computing the successive covering numbers, the CHAINING-UCB algorithm calibrates automatically the exploration-exploitation tradeoff. This fact contrasts with previous algorithms like GP-UCB where the input parameter β_t depends either on the cardinality $|\mathcal{X}|$ or on the Lipschitz constants of the kernel (see Theorems 1 and 2 of [9]). On the computational side, the discrete sum limits the number of ϵ_i-covers which need to be computed to only the ϵ_i such that $\epsilon_i > \min_{x\in\mathcal{X}} \sigma_n(x)$. Thanks to their geometrical decay, these numbers remain low in practice. Figure 1 illustrates the exploration/exploitation tradeoff we obtain with this discrete sum on a 1D toy example, compared to the tradeoff computed with a union bound as in GP-UCB. In Figure 1 a constant term is subtracted from the UCB in order to set the minimum of the exploration terms to zero in both CHAINING-UCB and GP-UCB.



(a) SE kernel (b) Himmelblau (c) Graph kernel

Figure 2: Empirical mean of the simple regret S_n in terms of iteration n for CHAINING-UCB, GP-UCB and RANDOM search (lower is better).

4 Practical considerations and experiments

Computing the ϵ-covers efficiently. As mentioned previously the computation of an optimal ϵ-cover is NP-hard. We demonstrate here how to build in practice a near-optimal ϵ-cover using a greedy algorithm on graph. First, remark that for any fixed ϵ we can define a graph $\mathcal{G}$ where the nodes are the elements of $\mathcal{X}$ and there is an edge between x and x' if and only if $d(x, x') \leq \epsilon$. The size of this construction is $\mathcal{O}(|\mathcal{X}|^2)$. The sparse structure of the underlying graph can be exploited to get an efficient representation. The problem of finding an optimal ϵ-cover reduces to the problem of finding a minimal dominating set on $\mathcal{G}$. We can therefore use the greedy Algorithm 2 which enjoys an approximation factor of $\log d_{\max}(\mathcal{G})$, where $d_{\max}(\mathcal{G})$ is the maximum degree of $\mathcal{G}$ (see for example [21] for a proof of NP-hardness and approximation results). This construction leads to an additional (almost constant) term of $\sqrt{\log\log d_{\max}(\mathcal{G})}$ in the right-hand side of Theorem 1. Finally, note that this approximation is optimal unless $\mathsf{P} = \mathsf{NP}$ as shown in [22].

Experiments. In this section we compare the ability of the CHAINING-UCB algorithm to find the maximum of an unknown function against the GP-UCB algorithm from [9] and the RANDOM search. For both the CHAINING-UCB and the GP-UCB algorithms the value for δ was set to 0.05. The RANDOM approach selects the next query uniformly among the unknown locations. It gives a baseline to grasp the scale of the performances of both algorithms. All three strategies are initialized with a set of 10 noisy observations sampled uniformly over $\mathcal{X}$. Figure 2 shows the empirical mean of the simple regret S_n over 32 runs. In every experiments the standard deviation of the noise was set to 0.05 and the search space is discrete with $|\mathcal{X}| = 10^4$. The SE experiment consists in generating GPs drawn from a two dimensional isotropic SE kernel, with the kernel bandwidth set to 1. The search space is a uniform design in a square of length 20. The Himmelblau experiment is a two dimensional polynomial function based on the Himmelblau's function with the addition of a linear trend. It possesses several local maxima which makes the global optimization challenging. The kernel used by the algorithm is an isotropic SE kernel with the bandwidth chosen by maximizing the marginal likelihood of a subset of training points. Finally we consider the task of optimizing over graphs. Global optimization of graphs can model complex problems as in industrial design, network analysis or computational biology. The input space is the set of directed graphs with less than 20 nodes. The kernel is the shortest-path kernel [23] normalized such that $k(g, g) = 1$ for $g \in \mathcal{X}$. Note that in this synthetic assessment we do not address the question faced in practice of choosing the prior. We further mention that the kernel matrix can be efficiently computed by pre-processing all pairs of shortest paths for each graph with Floyd-Warshall's algorithm. Figure 2 shows that the CHAINING-UCB algorithm is empirically more efficient than the GP-UCB algorithm on the three test cases. We remark that in practice, unlike GP-UCB, we may use a design with $|\mathcal{X}| \gg 10^4$ without affecting the performance of CHAINING-UCB. However generating a GP costs $\mathcal{O}(|\mathcal{X}|^3)$ which limits the tractability of synthetic assessments.

Conclusion. The theorem we derived and the experiment we performed suggest that the automatic calibration of the exploration/exploitation tradeoff by the hierarchical ϵ-covers effectively adapts to various settings. This chaining approach is a promising step toward generic, sound and tractable algorithms for Bayesian optimization.

References

[1] J. Snoek, H. Larochelle, and R. P. Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems 25*, pages 2960–2968, 2012.

[2] J. Mockus. *Bayesian approach to global optimization*. Mathematics and its applications. Kluwer Academic, 1989.

[3] D. R. Jones, M. Schonlau, and W. J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, December 1998.

[4] M. Osborne. *Bayesian Gaussian processes for sequential prediction, optimisation and quadrature*. PhD thesis, Oxford University New College, 2010.

[5] P. Hennig and C. J. Schuler. Entropy search for information-efficient global optimization. *Journal of Machine Learning Research*, 13:1809–1837, 2012.

[6] E. Contal, V. Perchet, and N. Vayatis. Gaussian process optimization with mutual information. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 253–261, 2014.

[7] C. E. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[8] A. Krause and C.S. Ong. Contextual Gaussian process bandit optimization. In *Advances in Neural Information Processing Systems 24*, pages 2447–2455, 2011.

[9] N. Srinivas, A. Krause, S. Kakade, and M. Seeger. Information-theoretic regret bounds for Gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, 2012.

[10] N. de Freitas, A. J. Smola, and M. Zoghi. Exponential regret bounds for Gaussian process bandits with deterministic observations. In *Proceedings of the 29th International Conference on Machine Learning*. icml.cc / Omnipress, 2012.

[11] E. Contal, D. Buffoni, A. Robicquet, and N. Vayatis. Parallel Gaussian process optimization with upper confidence bound and pure exploration. In *Machine Learning and Knowledge Discovery in Databases (ECML)*, volume 8188, pages 225–240. Springer Berlin Heidelberg, 2013.

[12] J. Djolonga, A. Krause, and V. Cevher. High-dimensional gaussian process bandits. In *Advances in Neural Information Processing Systems*, pages 1025–1033, 2013.

[13] S. Grunewalder, J-Y. Audibert, M. Opper, and J. Shawe-Taylor. Regret bounds for Gaussian process bandit problems. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 273–280. MIT Press, 2010.

[14] P. Gaillard and S. Gerchinovitz. A chaining algorithm for online nonparametric regression. *Proceedings of the Conference on Learning Theory (COLT)*, 2015.

[15] S. Boucheron, G. Lugosi, and P. Massart. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford University Press, 2013.

[16] E. Giné and R. Nickl. Mathematical foundations of infinite-dimensional statistical models, 2015.

[17] M. Talagrand. *Upper and Lower Bounds for Stochastic Processes: Modern Methods and Classical Problems*, volume 60. Springer-Verlag Berlin Heidelberg, 2014.

[18] D. Pollard. Empirical processes: Theory and applications. *NSF-CBMS Regional Conference Series in Probability and Statistics*, 2:pp. i–iii+v+vii–viii+1–86, 1990.

[19] R.M. Dudley. The sizes of compact subsets of hilbert space and continuity of gaussian processes. *Journal of Functional Analysis*, 1(3):290–330, 1967.

[20] E. Contal, C. Malherbe, and N. Vayatis. Optimization for gaussian processes via chaining. http://arxiv.org/abs/1510.05576, 2015.

[21] David S Johnson. Approximation algorithms for combinatorial problems. In *Proceedings of the fifth annual ACM symposium on Theory of computing*, pages 38–49. ACM, 1973.

[22] R. Raz and S. Safra. A sub-constant error-probability low-degree test, and a sub-constant error-probability pcp characterization of np. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 475–484. ACM, 1997.

[23] Karsten M Borgwardt and Hans-Peter Kriegel. Shortest-path kernels on graphs. In *Data Mining, Fifth IEEE International Conference on*, pages 8–pp. IEEE, 2005.